

Express Mail No. EL844343275US

PATENT APPLICATION OF
ERIC D. BRILL AND ARUL A. MENEZES
ENTITLED
METHOD AND APPARATUS FOR UNSUPERVISED
TRAINING OF NATURAL LANGUAGE PROCESSING
UNITS

FOI b7D b7E b7F b7G b7H b7I b7J b7K b7L b7M b7N b7O b7P b7Q b7R b7S b7T b7U b7V b7W b7X b7Y b7Z

Docket No. M61.12-0346

Method and Apparatus for Unsupervised Training of Natural Language Processing Units

BACKGROUND OF THE INVENTION

5 Natural language understanding involves
converting a string of characters into a meaning set
representing the meaning of the string of characters.
Such processing can involve a number of natural
language components including a segmentation
10 component that assigns characters to individual
words, a part of speech tagger that identifies the
part of speech of each word, a syntactic parser that
assigns a structure to a sentence or group of
sentences so that the syntactic relationship between
15 the words can be understood and a semantic
interpreter that analyzes the syntactic parse to
produce a semantic structure.

Each component in a natural language system
must be trained before it can be used. In the past,
20 such training has largely been done by hand. For
example, the rules used by syntactic parsers to parse
sentences were derived by hand. However, training by
hand is a laborious process of trial and error.
Because of this, more recent systems have attempted
25 to develop natural language components automatically,
using supervised machine learning techniques for
training.

For example, in supervised training of a
parser, a corpus of input sentences is created that

00493.0040
T04050"EE864860

is annotated to indicate the syntactic structure of each sentence. Such annotated sentences are referred to as tree banks in the art. During training, proposed changes to the parsing rules, known as
5 candidate learning sets, are tested by repeatedly parsing the tree bank using a different candidate learning set for each parse. The candidate learning set that provides the best parse based on the annotations in the tree bank is then used to change
10 the parser rules.

One problem with using supervised training is that it is expensive and time-consuming. For example, tree banks are so expensive and time-consuming to create that there are very few in
15 existence in the world.

Thus, a less expensive and less time-consuming method is needed for training natural language processing components.

SUMMARY OF THE INVENTION

20 A method of training a natural language processing unit applies a candidate learning set to at least one component of the natural language unit. The natural language unit is then used to generate a meaning set from a first corpus. A second meaning
25 set is generated from a second corpus using a second natural language unit and the two meaning sets are compared to each other to form a score for the candidate learning set. This score is used to determine whether to modify the natural language unit
30 based on the candidate learning set.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a general computing environment in which embodiments of the present invention may be practiced.

5 FIG. 2 is a flow diagram of a method of training a natural language unit under one embodiment of the present invention.

FIG. 3 is a block diagram of a training system under one embodiment of the present invention.

10 DETAILED DESCRIPTION OF ILLUSTRATIVE EMBODIMENTS

FIG. 1 illustrates an example of a suitable computing system environment 100 on which the invention may be implemented. The computing system environment 100 is only one example of a suitable
15 computing environment and is not intended to suggest any limitation as to the scope of use or functionality of the invention. Neither should the computing environment 100 be interpreted as having any dependency or requirement relating to any one or
20 combination of components illustrated in the exemplary operating environment 100.

The invention is operational with numerous other general purpose or special purpose computing system environments or configurations. Examples of
25 well known computing systems, environments, and/or configurations that may be suitable for use with the invention include, but are not limited to, personal computers, server computers, hand-held or laptop devices, multiprocessor systems, microprocessor-based
30 systems, set top boxes, programmable consumer

094933-050401
FOI-050-EEB64860

electronics, network PCs, minicomputers, mainframe computers, telephony systems, distributed computing environments that include any of the above systems or devices, and the like.

5 The invention may be described in the general context of computer-executable instructions, such as program modules, being executed by a computer. Generally, program modules include routines, programs, objects, components, data
10 structures, etc. that perform particular tasks or implement particular abstract data types. The invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing
15 environment, program modules may be located in both local and remote computer storage media including memory storage devices.

 With reference to FIG. 1, an exemplary
20 system for implementing the invention includes a general purpose computing device in the form of a computer 110. Components of computer 110 may include, but are not limited to, a processing unit 120, a system memory 130, and a system bus 121 that
25 couples various system components including the system memory to the processing unit 120. The system bus 121 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a
30 variety of bus architectures. By way of example, and

094933-0504-01
T04050-EE364360

not limitation, such architectures include Industry
Standard Architecture (ISA) bus, Micro Channel
Architecture (MCA) bus, Enhanced ISA (EISA) bus,
Video Electronics Standards Association (VESA) local
5 bus, and Peripheral Component Interconnect (PCI) bus
also known as Mezzanine bus.

Computer 110 typically includes a variety
of computer readable media. Computer readable media
can be any available media that can be accessed by
10 computer 110 and includes both volatile and
nonvolatile media, removable and non-removable media.
By way of example, and not limitation, computer
readable media may comprise computer storage media
and communication media. Computer storage media
15 includes both volatile and nonvolatile, removable and
non-removable media implemented in any method or
technology for storage of information such as
computer readable instructions, data structures,
program modules or other data. Computer storage
20 media includes, but is not limited to, RAM, ROM,
EEPROM, flash memory or other memory technology, CD-
ROM, digital versatile disks (DVD) or other optical
disk storage, magnetic cassettes, magnetic tape,
magnetic disk storage or other magnetic storage
25 devices, or any other medium which can be used to
store the desired information and which can be
accessed by computer 110. Communication media
typically embodies computer readable instructions,
data structures, program modules or other data in a
30 modulated data signal such as a carrier wave or other

09849833-050404

The system memory 130 includes computer storage media in the form of volatile and/or nonvolatile memory such as read only memory (ROM) 131 and random access memory (RAM) 132. A basic input/output system 133 (BIOS), containing the basic routines that help to transfer information between elements within computer 110, such as during start-up, is typically stored in ROM 131. RAM 132 typically contains data and/or program modules that are immediately accessible to and/or presently being operated on by processing unit 120. By way of example, and not limitation, FIG. 1 illustrates operating system 134, application programs 135, other program modules 136, and program data 137.

The computer 110 may also include other removable/non-removable volatile/nonvolatile computer storage media. By way of example only, FIG. 1 illustrates a hard disk drive 141 that reads from or
30 writes to non-removable, nonvolatile magnetic media,

104050" EE864860

a magnetic disk drive 151 that reads from or writes to a removable, nonvolatile magnetic disk 152, and an optical disk drive 155 that reads from or writes to a removable, nonvolatile optical disk 156 such as a CD ROM or other optical media. Other removable/non-removable, volatile/nonvolatile computer storage media that can be used in the exemplary operating environment include, but are not limited to, magnetic tape cassettes, flash memory cards, digital versatile disks, digital video tape, solid state RAM, solid state ROM, and the like. The hard disk drive 141 is typically connected to the system bus 121 through a non-removable memory interface such as interface 140, and magnetic disk drive 151 and optical disk drive 155 are typically connected to the system bus 121 by a removable memory interface, such as interface 150.

The drives and their associated computer storage media discussed above and illustrated in FIG. 1, provide storage of computer readable instructions, data structures, program modules and other data for the computer 110. In FIG. 1, for example, hard disk drive 141 is illustrated as storing operating system 144, application programs 145, other program modules 146, and program data 147. Note that these components can either be the same as or different from operating system 134, application programs 135, other program modules 136, and program data 137. Operating system 144, application programs 145, other program modules 146, and program data 147 are given

different numbers here to illustrate that, at a minimum, they are different copies.

A user may enter commands and information into the computer 110 through input devices such as a keyboard 162, a microphone 163, and a pointing device 161, such as a mouse, trackball or touch pad. Other input devices (not shown) may include a joystick, game pad, satellite dish, scanner, or the like. These and other input devices are often connected to the processing unit 120 through a user input interface 160 that is coupled to the system bus, but may be connected by other interface and bus structures, such as a parallel port, game port or a universal serial bus (USB). A monitor 191 or other type of display device is also connected to the system bus 121 via an interface, such as a video interface 190. In addition to the monitor, computers may also include other peripheral output devices such as speakers 197 and printer 196, which may be connected through an output peripheral interface 190.

The computer 110 may operate in a networked environment using logical connections to one or more remote computers, such as a remote computer 180. The remote computer 180 may be a personal computer, a hand-held device, a server, a router, a network PC, a peer device or other common network node, and typically includes many or all of the elements described above relative to the computer 110. The logical connections depicted in FIG. 1 include a local area network (LAN) 171 and a wide area network

(WAN) 173, but may also include other networks. Such networking environments are commonplace in offices, enterprise-wide computer networks, intranets and the Internet.

5 When used in a LAN networking environment, the computer 110 is connected to the LAN 171 through a network interface or adapter 170. When used in a WAN networking environment, the computer 110 typically includes a modem 172 or other means for
10 establishing communications over the WAN 173, such as the Internet. The modem 172, which may be internal or external, may be connected to the system bus 121 via the user input interface 160, or other appropriate mechanism. In a networked environment,
15 program modules depicted relative to the computer 110, or portions thereof, may be stored in the remote memory storage device. By way of example, and not limitation, FIG. 1 illustrates remote application programs 185 as residing on remote computer 180. It
20 will be appreciated that the network connections shown are exemplary and other means of establishing a communications link between the computers may be used.

25 The present invention provides a method and apparatus for performing unsupervised training of one or more natural language processing components, such as syntactic parsers and/or semantic interpreters. The invention performs this training by utilizing at least two natural language processing systems,
30 typically consisting of a syntactic parser and

semantic interpreter, possibly with other components. These systems are used to form separate meaning sets from parallel corpora, which represent the same set of sentences written in different languages. Thus, 5 one natural language processing system generates a meaning set from the corpus written in a first language, such as English, while a second natural language processing system generates a meaning set from the corpus written in a second language, such as 10 French. The two meaning sets are compared and the configuration of one or more of the components of the natural language processing systems is adjusted so that the meaning sets converge toward each other. The configurations for the components that provide 15 the best convergence for the meaning sets are then selected and the training ends.

The method and apparatus of the present invention are described below with reference to the flow diagram of FIG. 2 and the block diagram of FIG. 20 3.

In the block diagram of FIG. 3, two sets of natural language processing units 300 and 302 are provided. Each natural language processing unit derives a meaning set from a corpus written in a 25 separate language. Thus, natural language processing unit 300 derives a meaning set 304 from a corpus 306 written in a language "S", while natural language processing unit 302 derives a meaning set 308 from a corpus 310 written in a language "E". Note that

language "S" and language "E" can be any known written language.

Language S corpus 306 and language E corpus 308 contain sentences expressing identical meaning written in different languages under most embodiments. Thus, together they form a single bilingual corpus. The sentences in each corpus are aligned such that a sentence or groups of sentences that convey a meaning in one corpus are aligned with the sentences or groups of sentences that convey the same meaning in the other corpus. For instance, if the meaning in the first sentence in corpus 306 is the same as the meaning in the first sentence in corpus 308, the first sentence in corpus 306 is aligned with the first sentence in corpus 308. Note that if the meaning found in one sentence of a corpus is expressed in two sentences in the other corpus, the single sentence of the first corpus would be aligned with the two sentences of the second corpus.

Natural language processing unit 300 includes a syntactic parser 320 and a semantic interpreter 322 for language "S". The operation of syntactic parser 320 and semantic interpreter 322 are controlled by a parser specification 324 and an interpreter specification set 326, respectively, where each specification defines the input/output mappings of the component. For example, the interpreter specification defines the meaning sets that will be produced at the output of the interpreter for various input semantic structures.

Similarly, natural language processing unit 302 includes a language "E" syntactic parser 330 and a semantic interpreter 332 that are controlled by a parser specification 334 and an interpreter specification 336, respectively.

Note that natural language processing units 300 and 302 can also include other natural language components such as a part of speech tagger or a segmentation component. Although these additional components are not shown in FIG. 3 for simplicity, those skilled in the art will recognize that the present invention can be applied to any of the trainable natural language components that are present in a natural language understanding unit.

Under the method of the present invention, one or more of the specifications 324, 326, 334 and/or 336 are adjusted through unsupervised training. In the description below, an unsupervised training method involving generating and testing candidate learning sets is described. However, those skilled in the art will recognize that the present invention may be incorporated in other unsupervised training techniques such as greedy hill climbing and variants of the expectation-maximization algorithm.

The generating and testing embodiment of the present invention is shown in FIG. 2 and begins at step 200 where an unsupervised learning module 352 in FIG. 3 selects an initial specification for each of the parsers and interpreters. In addition, learning module 352 selects one or more learning

sets, which are candidate changes that are to be tested using the process of FIG. 2. Specifically, each learning set will be individually applied to the various natural language component specifications.

- 5 By applying each learning set, the training method of the present invention is able to determine which candidate change provides the best improvement in the natural language units.

Using the selected specifications, the
10 process of FIG. 2 continues at step 201 where natural language processing unit 300 performs natural language processing on language "S" corpus 306 to produce a baseline meaning set 304. At step 202, natural language processing unit 302 performs natural
15 language processing on language "E" corpus 308 to produce a baseline meaning set 310.

Once each natural language processing unit has formed its meaning sets, the meaning sets are compared to one another at step 204 by a score
20 calculator 350 to generate a combined score for the two meaning sets. Under one embodiment, this score is generated using the following distance calculation:

$$\text{MR Distance} = \frac{\sum_{i=1}^N \text{Sim}(M_i^E, M_i^S)}{N} \quad \text{EQ. 1}$$

25 where $\text{Sim}(M_i^E, M_i^S)$ is a similarity function that provides some measure of the similarity between a meaning M_i^E for a sentence "i" in language corpus E and a meaning M_i^S for corresponding sentence or

sentences i in language corpus S . In Equation 1, the sum is performed over all N sentences in each corpus, with the division providing an average similarity or distance score.

5 After the score for the current component specifications have been determined, unsupervised learning module 352 determines whether there is a learning set that should be tested at step 206. If there is a learning set that needs to be tested, the
10 process continues at step 208 where learning module 252 changes one or more of the component specifications to implement the learning set to be tested. The process then repeats steps 201, 202 and 204 by processing the bilingual language corpus using
15 the current learning set and comparing the resulting meaning sets to produce a score for the current learning set.

 Steps 201, 202, 204, 206 and 208 are repeated until a score has been generated for each
20 candidate learning set. When there are no more learning sets to be tested at step 206, the process continues at step 210 where the learning set that provides the highest score is selected. The appropriate component specifications are then changed
25 to implement the selected learning set.

 The process of FIG. 2 can be repeated for multiple learning sets, thereby progressively improving the component specifications of the natural language units.

004050-EE864860

Note that although two natural language units were used in FIGS. 2 and 3, in other embodiments, larger numbers of natural language units are used. In such embodiments, the learning sets can
5 be limited to a single natural language unit or may be distributed across several natural language units.

In embodiments that limit the learning sets to a single natural language unit, a separate distance measure is determined between the changing
10 natural language unit and all of the other natural language units. Under one embodiment, these separate distance scores are averaged to form a single distance score for the current learning set. In other embodiments, the maximum or minimum distance
15 score is selected as the distance score for the learning set.

Because the present invention adjusts each natural language component specification based on the meaning set provided by the natural language unit,
20 each component is improved so that it provides a better meaning set instead of just a better output from the component itself. Since producing a good meaning set is goal of a natural language understanding unit, it is thought that improving each
25 component based on the output meaning set produces more useful components for natural language processing units.

Note that once a natural language component has been trained under the present invention, it may
30 be used outside of natural language understanding.

Thus, a syntactic parser trained under the present invention may be used in a grammar checker that does not include a semantic interpreter.

Although the invention above was described
5 in terms of generating a meaning set, in other
embodiments, the natural language unit generates a
set of actions based on the language corpus instead
of a meaning set. For example, in an embodiment
where the natural language unit is used to control a
10 robot, the output of the natural language unit is a
set of actions performed by the robot. In such
embodiments, the set of actions produced by natural
language units 300 and 302 are compared to each other
to generate a score instead of using a meaning set.
15 This score is then used to modify one or more of the
natural language units so that the actions produced
by the two units are more similar.

Although the present invention has been
described with reference to particular embodiments,
20 workers skilled in the art will recognize that
changes may be made in form and detail without
departing from the spirit and scope of the invention.

FOIA b 7 - EE 864860